OIPE JC44
APR 22 2002
PATENT & TRADEMARK OFFICE

03CO
#9

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	§	Attorney Docket No. **RPS920000119US1**
DEAN ALAN KALMAN	§	
Serial No. **09/894,077**	§	Examiner: **UNKNOWN**
Filed: **June 28, 2001**	§	
Title: **SYSTEM AND METHOD FOR SERVICING REQUESTS TO A STORAGE ARRAY**	§	Art Unit: **2184**

COPY OF PAPERS ORIGINALLY FILED

PETITION TO WITHDRAW ERRONEOUS NOTICE OF OMITTED ITEM(S)

RECEIVED
MAY 2 0 2002
OFFICE OF PETITIONS

Assistant Commissioner of Patents
Washington, DC 20231

Sir:

In response to the Notice of Omitted Item(s) dated March 11, 2002, Applicant respectfully traverses the Office's contention that Figures 3A and 3B have been omitted from the application and avers that Figures 3A and 3B were, in fact, deposited with the PTO with the original application papers. In support of Applicant's position, Applicant supplies herewith a copy of the application originally filed, together with a return postcard bearing the U.S. PTO stamp. This return receipt postcard demonstrates that Figures 3A and 3B which are found on the third and fourth sheets of drawings, respectively, were not omitted from the application as filed.

CERTIFICATE OF MAILING
37 CFR 1.8(a)

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as First Class Mail in an envelope addressed to: Assistant Commissioner for Patents; Box: OIPE; Washington, D. C. 20231 on 4/11, 2002.

Kim DeLeon
Typed or printed name of person signing paper

Signature of person signing paper

04/24/2002 BNGUYEN1 00000051 500563 09894077
01 FC:122 130.00 CH

Applicant authorizes a charge of $130.00 for the petition fee specified by 37 C.F.R. §1.17(h) to **IBM Corporation Deposit Account No. 50-0563**. No additional fee is believed to be required; however, if any additional fees are required for the prosecution of this application, please charge such fees to the above-identified Deposit Account No.

Respectfully submitted,

Brian F. Russell

Brian F. Russell
Registration No. 40,796
BRACEWELL & PATTERSON, L.L.P.
Suite 350, Lakewood on the Park
7600B North Capital of Texas Highway
Austin, TX 78731
(512) 343-6116
ATTORNEY FOR APPLICANT

RECEIVED
MAY 2 0 2002
OFFICE OF PETITIONS

OIPE JC44
APR 22 2002
PATENT & TRADEMARK OFFICE

THE ASSISTANT COMMISSIONER OF PATENTS
Washington, D.C. 20231

DOCKET NUMBER: RPS920000119US1
Date: June 28, 2001

Sir:

Transmitted herewith for filing is the Patent Application of:

Inventor: Dean Alan Kalman

For: **System and Method for Servicing Requests to a Storage Array**

COPY OF PAPERS
ORIGINALLY FILED

Enclosed are:

X Patent Specification

X Drawings - Five (5) sheets

X Declaration and Power of Attorney

X An assignment of the invention to International Business Machines Corporation (includes Recordation Form Cover Sheet).

X Return Postcard

__ A certified copy of a _____ application.

__ Information Disclosure Statement, PTO 1449 and copies of references.

The filing fee has been calculated as shown below:

For	Number Filed		Number Extra		Rate		Fee
Basic Fee							$ 710.00
Total Claims	26	- 20	6	X	$ 18.00	=	$ 108.00
Independent. Claims	4	- 3	1	X	$ 80.00	=	$ 80.00
MULTIPLE DEPENDENT CLAIM PRESENTED			0	X	$ 270.00	=	$ 0.00
					TOTAL		$ 898.00

X Please charge IBM Corporation Deposit Account No. 50-0563 in the amount of $898.00. A duplicate copy of this sheet is enclosed.

X The Commissioner is hereby authorized to charge payment of the following fees associated with this communication or credit any overpayment to IBM Corporation Deposit Account 50-0563. A duplicate copy of this sheet is enclosed.

CERTIFICATE OF MAILING BY "EXPRESS MAIL" UNDER 37 CFR § 1.10

"Express Mail" mailing label number: EF 272757360US

Date of Mailing: June 28, 2001

I hereby certify that the documents indicated below are being deposited with the United States Postal Service under 37 CFR 1.10 on the date indicated above and are addressed to Box Patent Applications, Assistant Commissioner of Patents, Washington, D.C. 20231 and mailed on the above Date of Mailing with the above "Express Mail" mailing label number.

Dena Faris
(name of person mailing paper)

Dena Faris
SIGNATURE of person mailing paper or fee

Respectfully submitted,

By
Andrew J. Dillon
Registration No. 29,634
BRACEWELL & PATTERSON, LLP
Suite 350 Lakewood on the Park
7600B North Capital of Texas Highway
Austin, Texas 78731
Telephone (512) 343-6116

OIPE JC44
APR 22 2002
PATENT & TRADEMARK OFFICE

Recycled Paper

EXPRESS MAIL

UNITED STATES POSTAL SERVICE®

www.usps.com

AFFIX POSTAGE O.
CORPORATE ACC:
ADHIERA AQUÍ EL
ETIQUETA DE CUE
USO NACIONAL Ú.

POST OFFICE TO ADDRESSEE

EXPRESS MAIL
UNITED STATES POSTAL SERVICE TM

EF272757360US

Mailing Label
Label 11-B May 2000

ORIGIN (POSTAL USE ONLY)			
PO ZIP Code	Day of Delivery ☐ Next ☐ Second	Flat Rate Envelope ☐	
Date In Mo. Day Year	☐ 12 Noon ☐ 3 PM	Postage $	
Time In ☐ AM ☐ PM	Military ☐ 2nd Day ☐ 3rd Day	Return Receipt Fee	
Weight lbs. ozs.	Int'l Alpha Country Code	COD Fee	Insurance Fee
No Delivery ☐ Weekend ☐ Holiday	Acceptance Clerk Initials	Total Postage & Fees $	

DELIVERY (POSTAL USE ONLY)		
Delivery Attempt Mo. Day	Time ☐ AM ☐ PM	Employee Signature
Delivery Attempt Mo. Day	Time ☐ AM ☐ PM	Employee Signature
Delivery Date Mo. Day	Time ☐ AM ☐ PM	Employee Signature
Signature of Addressee or Agent X		
Name-Please Print X		

CUSTOMER USE ONLY

METHOD OF PAYMENT
Express Mail Corporate Acct. No.
Federal Agency Acct. No. or Postal Service Acct. No.

☐ WAIVER OF SIGNATURE (Domestic Only) I wish delivery to be made without obtaining signature of addressee or addressee's agent (if delivery employee judges that article can be left in secure location) and I authorize that delivery employee's signature constitutes valid proof of delivery.
NO DELIVERY ☐ Weekend ☐ Holiday
Customer Signature

FROM: (PLEASE PRINT) PHONE (512) 343-6116

Bracewell & Patterson L.L.P.
Suite 350 Lakewood on the Park
7600B N. Capital of TX Highway
Austin, TX 78731

0252AP.045472

TO: (PLEASE PRINT) PHONE ( )

HON. Comm. of Patents
U.S. Patent + Trademark Of
Washington, D.C. 20231
Box: New Patent Applica

PRESS HARD.
You are making 3 copies.

FOR PICKUP OR TRACKING CALL 1-800-222-1811 www.usps.com

EMS

P-13A SEPTEMBER 1999

the above "Express Mail" mailing label number.

Dena Faris
(name of person mailing paper)

Dena Faris
SIGNATURE of person mailing paper or fee

BRACEWELL & PATT
Suite 350 Lakewood or
7600B North Capital c
Austin, Texas 78731
Telephone (512) 343

#11

a. client 6/28

COPY OF PAPERS
ORIGINALLY FILED

Received in the U.S.P.T.O.
In re Application of:
DEAN ALAN KALMAN
SYSTEM AND METHOD FOR SERVICING REQUESTS TO A STORAGE ARRAY
ENCLOSED HEREWITH:

1. *Transmittal of Patent Application (in duplicate) w/ Certificate of Express Mail No.EF272757360US ;*
2. *Subject Patent Application containing 12 pgs. of description, 7 pgs. of claims and 1 pg. Abstract;*
3. *Executed Declaration and Power of Attorney;*
4. *Form PTO-1595 Recordation Cover Sheet;*
5. *Assignment;*
6. *Five (5) sheets of drawings;and*
7. *Our return postcard.*

IBM CORPORATION
Our File No. 0252AD-RPS920000119US1
Mailed: ***June 28, 2001***

7/13

Received in the U.S.P.T.O.
In re Application of:
DEAN ALAN KALMAN
SYSTEM AND METHOD FOR SERVICING REQUESTS TO A STORAGE ARRAY
ENCLOSED HEREWITH:

1. *Transmittal of Patent Application (in duplicate) w/ Certificate of Express Mail No.EF272757360US ;*
2. *Subject Patent Application containing 12 pgs. of description, 7 pgs. of claims and 1 pg. Abstract;*
3. *Executed Declaration and Power of Attorney;*
4. *Form PTO-1595 Recordation Cover Sheet;*
5. *Assignment;*
6. *Five (5) sheets of drawings;and*
7. *Our return postcard.*

IBM CORPORATION
Our File No. 0252AD-RPS920000119US1
Mailed: ***June 28, 2001***

OIPE JC44 APR 2 2 2002 PATENT & TRADEMARK OFFICE

COPY OF PAPERS ORIGINALLY FILED

SYSTEM AND METHOD FOR SERVICING REQUESTS TO A STORAGE ARRAY

BACKGROUND OF THE INVENTION

1. Technical Field:

The present invention relates in general to data processing systems, and more particularly, data storage within data processing systems. Still more particularly, the present invention relates to a system and method for handling data requests within data processing systems.

2. Description of the Related Art:

In modern computer systems, users frequently access common data stored in a redundant array of inexpensive disks (RAID) of a central networked computer. Redundancy is employed so that if a hard disk of the RAID fails, the data stored on the failed hard disk can be rebuilt utilizing data and parity data stored on the other hard disks of the RAID while the failed hard disk is taken off-line and repaired.

Hard disks within a RAID frequently are subject to numerous accesses ranging from relatively short data writes or reads to relatively lengthy updates of the configuration of the hard disk. During access to a disk, one or more users of the computer system may be unable to access the data on the disk. Such users typically wait until the hard disk completes the previous process until their data access requests can be fulfilled.

An example of a lengthy hard disk update that would restrict user access to data is a firmware update. Firmware updates may be required, for example, to resolve incompatibilities between components (e.g., hard disks) or to correct errors in the firmware that are not detected until after the release of the firmware.

An update to the firmware of a conventional hard disk cannot be performed when the drive is on-line. Because of this limitation, hard disk firmware updates are performed in a controlled operating system environment with the selected hard disk set to an off-line mode to prevent access to the hard disk. Utilizing a single threaded environment (e.g., disk operating system, or DOS) allows a hard disk firmware update application to assume control of the computer system and prevent other applications from accessing the hard disk during the firmware update.

Because many computer systems and networks employ multi-threaded operating systems (e.g., Windows NT), to perform a firmware update, an administrator typically shuts down the multi-threaded operating system, takes the computer system off-line, boots DOS, runs the hard disk update software, and waits for the update to complete. During the firmware update, the RAID is taken off-line, which prevents other users from accessing the data stored on the RAID.

Thus, it is often the case that users may request access to data stored on a hard disk that is unavailable due to an update, repair, or another access. The present invention recognizes that a system and method for servicing a data request to an unavailable hard disk without waiting for the update, access, or repair to be completed, would improve the overall of a data processing system.

SUMMARY OF THE INVENTION

To overcome the foregoing and additional limitations of the prior art, the present invention presents an improved system and method of servicing a data request to a storage media array. In accordance with a preferred embodiment of the present invention, the data processing system includes an adapter and a redundant array of inexpensive disks (RAID). The adapter includes an updating circuit and a request servicing circuit.

According to the method of the present invention, if a data read request to an unavailable particular storage medium is received from a system processor, a request servicing circuit of an adapter reconstructs the request data from data and parity data stored on other storage media in the array. However, if a data write request to an unavailable storage medium is received from the system processor, the actual data related to the data write request is stored in a temporary memory location, and the data write request is logged in a non-volatile memory. There is no built-in data redundancy in a first preferred embodiment of the present invention; thus, modified data is lost in the event of the failure of another storage medium during the update of the selected storage medium. However, a second preferred embodiment of the present invention incorporates a spare storage medium that preserves the modified data if another storage medium fails.

BRIEF DESCRIPTION OF THE DRAWINGS

The novel features believed characteristic of the invention are set forth in the appended claims. The invention itself however, as well as a preferred mode of use, further objects and advantages thereof, will best be understood by reference to the following detailed description of an illustrative embodiment when read in conjunction with the accompanying drawings, wherein:

Figure 1 depicts a detailed block diagram of an exemplary data processing system, which may be utilized to implement the present invention;

Figure 2 illustrates a detailed block diagram of an adapter for a storage array in accordance with a preferred embodiment of the present invention;

Figure 3A depicts a high-level logic flowchart illustrating a first method for servicing a data request to a storage array according to a preferred embodiment of the present invention;

Figure 3B illustrates a table depicting a redundant array of inexpensive disks (RAID) implementing a first method of servicing a data request to a storage array according to a preferred embodiment of the present invention;

Figure 4A depicts a high-level logic flowchart illustrating a second method for servicing a data request to a storage array including a spare storage medium according to a preferred embodiment of the present invention; and

Figure 4B illustrates a table depicting a RAID implementing a second method of servicing a data request to a storage array according to a preferred embodiment of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

With reference now to the figures and in particular with reference to **Figure 1**, there is depicted a detailed block diagram of a data processing system **10** in which a preferred embodiment of the present invention may be implemented. As illustrated, data processing system **10** includes a system processor **12**, a system memory **14**, a user interface **17**, and a communications control unit **19**, all coupled via a system interconnect **16**. System processor **12**, which may be implemented as one or more processors, such as any generation of Pentium™ processor available from Intel Corporation in Santa Clara, CA, executes software instructions to control the operation of data processing system **10**. System memory **14** provides storage for software and data and may include both read-only memory (ROM) and random access memory (RAM).

It should be readily apparent to those skilled in the art that system interconnect **16** may be implemented as a bus, a switch or any other type of coupling apparatus. User interface **17** provides an interface between data processing system **10** and peripherals employed by a user to interact with data processing system **10**. User interface **17** may include various adapters and drivers for controlling peripherals, such as keyboard **28**, display **30**, and mouse **32**. Communications control unit **19** provides an interface between data processing system **10** and an external network **26**, which may be a local area network (LAN) or a wide area network (WAN) such as the Internet.

An adapter **22** controls access and updates to a storage media array **23** via an array interconnect **25** and is coupled to system processor **12** by a mezzanine interconnect **18**, which in a preferred embodiment of the present invention may be implemented as a peripheral component interconnect (PCI) bus. Array interconnect **25** can be implemented as a small computer system interface (SCSI) bus.

It should be readily apparent to those skilled in the art that any interconnect system (e.g., integrated device electronics, or IDE) may be implemented to couple storage media array **23** to system processor **12**. Storage media array **23** can be implemented as a redundant array of inexpensive disks (RAID) **24a-24n**.

Referring now to **Figure 2**, a detailed block diagram is illustrated of an adapter **22** utilized by a data processing system **10**, as depicted in **Figure 1**, according to a preferred embodiment of the present invention. Adapter **22** coordinates updates and services data requests to storage media array **23** by system processor **12** or by other networked data processing systems coupled to data processing system **10**.

As depicted, adapter **22** includes an adapter processor **104** that is implemented as an updating circuit for coordinating accesses and updates to storage media array **23**. Coupled to adapter processor **104** via an adapter interconnect **102** is an adapter processor memory **106**, which functions as a cache memory for adapter processor **104**.

Further coupled to adapter processor **104** is non-volatile random access memory (NVRAM) **108**. During an update to a selected storage medium of storage media array **23**, adapter **22** receives a data write request from system processor **12**. NVRAM **108** logs the data write request as a "dirty stripe", or a stripe of modified data. The actual data from the data write request are stored in another storage medium in the array. Requests for data during an update to the storage medium of storage media array **23** are handled by a request servicing circuit, implemented as an exclusive-OR (XOR) processor **112**, coupled by adapter interconnect **102**.

Adapter **22** finally includes PCI/PCI bridge **114**, and small computer system interface (SCSI) channels **116** to facilitate data communication between system PCI bus **18** and SCSI bus **25**.

With reference now to **Figure 3A**, there is illustrated a high-level logic flowchart depicting an exemplary method by which an adapter **22** services a request to access a storage media array **23** according to a first embodiment of the present invention. In **Figure 3B**, there is depicted a table that illustrates an exemplary RAID that stores data in stripes across three storage media.

As depicted in **Figure 3A**, the method begins at block **200** and then continues to an update loop including blocks **202**, **204**, and **206**. During the execution of the update loop, an adapter processor **104** (or other suitable updating circuit) selects a storage medium to update out of storage media array **23**, as shown in block **202**. In this example, adapter **22** selects storage medium **2** for updating, as depicted in **Figure 3B**. As illustrated in block **204**, adapter processor **104** sets the storage medium **2** to an off-line status, and begins relaying update data (e.g., updated firmware) to storage medium **2**, as depicted in block **206**.

The process then enters a request servicing loop that includes blocks **208**, **210**, **212**, **213**, **214**, **216**, **218** and **220**. The request servicing loop illustrates the process of providing a user with requested data that is located on a selected, inaccessible storage medium. In this example, as depicted in **Figure 3B**, storage medium **2** is inaccessible at this time because adapter **22** had been instructed by system processor **12** to perform an update to storage medium **2**.

During the update of storage medium **2**, adapter **22** monitors for data requests to the selected storage medium, as shown in block **208**. If adapter **22** monitors a data request to the selected storage medium and the data request is a data read request (e.g., a data read request for data **D1** of **Figure 3B**), as depicted in block **210**, the requested data **D1** are automatically reconstructed by a request servicing circuit, implemented as a XOR processor **112**, utilizing data **D0** and parity data **P1** stored in other storage media in the array, as shown in block **212**. The request servicing circuit

provides the requested data to the requesting component, as depicted in block **213**. The process then returns from block **213** to block **208**.

If the data request is not a data read request, the data request is assumed to be a data write request, as illustrated in block **214**. The data write request and the location of the data related to the data write request are logged within NVRAM **108**, and the actual data corresponding to the data write request are stored another storage medium in the array, as depicted in blocks **216** and **218**. The process then returns from block **218** to block **208**, where the process monitors for data requests to storage media **2**, as illustrated.

If adapter **22** does not receive a data request to the selected storage medium, the process moves from block **208** to block **220**, which depicts a determination of whether or not the update of the selected storage medium is complete. If the update of the selected storage medium is not complete, the process returns from block **220** to block **208** where adapter **22** continues to monitor for a data request to the selected storage medium.

If the update of the selected storage medium is complete, the process continues to block **222**, where each dirty stripe logged in NVRAM is rebuilt with data and parity data stored on other the storage media in the storage media array, and the new data stored in the temporary location in the other storage media. The selected storage medium is then set to an online status by adapter **22**, as depicted in block **224**. Finally, if there are remaining storage media to be updated, the process returns from block **226** to block **202**, where the process reenters the update loop. If there are no remaining storage media to be updated, the process ends, as shown in block **228**.

One drawback to the method outlined by a preferred first embodiment of the present invention, as depicted in **Figures 3A** and **3B**, is the lack of redundancy in the

system. If another storage medium (storage media 1 or 3, as illustrated in **Figure 3B**) fails during the update of storage medium 2, the entire storage media array 23 is taken off-line. The failed storage medium (storage media 1 or 3) is repaired and then rebuilt utilizing data and parity data from the other two storage media. However, any modified data for storage medium 2 temporarily stored on the failed storage medium during the update are lost.

Referring to **Figures 4A**, a high-level logic flowchart depicting an exemplary method by which an adapter 22 services a request to access a storage media array according to a second embodiment of the present invention. Illustrated in **Figure 4B**, there is depicted an exemplary RAID that stores data in stripes across four storage media. Included in storage media array 23 is a spare storage medium for storing data and write data requests during a selected storage medium update controlled by adapter 22.

As depicted, the method begins at block 300 and then continues to block 302, which is illustrated as an update loop including blocks 302, 304 and 306. The update loop depicts the selection and update of a storage medium. During the execution of the update loop, an adapter processor 104, implemented as an updating circuit, selects a storage medium to update out of a storage media array 23, as shown in block 302. In this example, according to **Figure 4B**, adapter 22 selects storage medium 2 for updating. As illustrated in block 304, adapter processor 104 sets storage medium 2 to an off-line status, and begins relaying update data to storage medium 2, as depicted in block 306.

The process then enters a request servicing loop that includes blocks 308, 310, 312, 313, 314, 316, 318 and 320. The request servicing loop illustrates the process of providing a user with requested data that is located on a selected, inaccessible storage medium. The selected storage medium is inaccessible

at this time because adapter **22** had been instructed by system processor **12** to perform an update on storage medium **2**.

During the update of storage medium **2**, adapter **22** monitors for data requests to storage medium **2**, as shown in block **308**. For example, according to **Figure 4B**, if adapter **22** monitors a data request determined to be a data read request for data **D1** to storage medium **2**, as depicted in block **310**, the requested data **D1** is automatically reconstructed by XOR processor **112**, (or other request servicing circuit), from data **D0** and parity data **P1** stored in other storage media (e.g., storage media **1** and **3**) in the array, as shown in block **312**. The request servicing circuit provides the requested data to the requesting component, as depicted in block **313**. The process then returns from block **313** to block **308**. If the data request is not a data read request, the data request is assumed to be a data write request to storage medium **2**, as illustrated in block **314**. The data write request is logged within a NVRAM **108** and a spare storage medium **4**, as illustrated in block **316**. The actual data **D1** associated with the data write request is stored in the spare storage medium, as data **D1'**, as depicted in block **318**. The process then returns to block **308** from block **318**.

If adapter **22** does not receive a data request to the selected storage medium, the process moves to block **320**, which depicts a determination of whether or not the update to the selected storage medium is complete. If the update of storage medium **2** is complete, the process continues to block **322**, where each dirty stripe logged in NVRAM is rebuilt with data and parity data stored in the other storage media in the array and the new data stored in spare storage medium **4**; as illustrated, storage medium **2** is set to an online status by adapter **22**, as depicted in block **324**. Finally, if there are remaining storage media to be updated, the process returns from block **326** to block **302**, where the process reenters the update loop. If there are no remaining storage media to be updated, the process ends, as shown in block **328**.

If the update of storage medium **2** is not complete, the process continues to block **330**, which illustrates adapter **22** checking if another storage medium of the array has failed. If another storage medium of the array has failed, storage media array **23** is set to off-line status by adapter **22**, as shown in block **332**. The update of storage medium **2** is completed, as depicted in block **334**, and storage medium **2** is rebuilt with data and parity data stored in other storage media, and new data stored in the spare storage medium **4**, as illustrated in block **336**. The process then continues to block **324**. If another storage medium of the array has not failed, the process returns from block **330** to block **308**.

This second preferred embodiment of the present invention addresses the drawback of the first embodiment. As illustrated in **Figure 4B**, spare storage medium **4** records the modified data to be transferred to storage medium **2** when the update is complete. Therefore, if one of the other storage media fails during the update of storage medium **2**, the modified data are preserved on spare storage medium **4**.

As described above, an improved system and method of servicing a request to an array of storage media is presented. A data processing system, according to a preferred embodiment of the present invention includes a storage media array and an adapter to interface with the storage media array. If a data read request to a particular storage medium is received from a system processor, a request servicing circuit of the adapter reconstructs the request data from data and parity data stored on other storage media in the array. However, if a data write request to an unavailable storage medium is received from the system processor, the actual data related to the data write request are stored in a temporary memory location, and the data write request is logged in a non-volatile random access memory (NVRAM). There is no built-in data redundancy in a first preferred embodiment of the present invention; thus, modified data are lost in the event of the failure of another storage medium during the update of the selected

storage medium. However, a second preferred embodiment of the present invention incorporates a spare storage medium that preserves the modified data if another storage medium fails.

While the invention has been particularly shown and described with reference to a preferred embodiment, it will be understood by those skilled in the art that various changes in form and detail may be made therein without departing from the spirit and scope of the invention. For example, the data on a storage medium in a RAID utilizing the present invention may still be accessed by a user even if the storage medium is off-line due to required maintenance.

CLAIMS

What is claimed is:

1. A method of servicing a request to a redundant storage array including a plurality of storage media containing data and related parity data, said method comprising:

receiving a data request for data stored by a selected storage medium among said plurality of storage media; and

in response to receiving said data request while an update to said selected storage medium is being performed, servicing said data request by reference to said data and said related parity data of other storage media among said plurality of storage media.

2. The method according to Claim 1, said step of servicing further including:

automatically reconstructing requested data utilizing said data and related parity data from said other storage media; and

outputting said reconstructed requested data.

3. The method according to Claim 2, said step of automatically reconstructing further including:

determining a type of said data request; and

automatically reconstructing requested data utilizing said data and said related

parity data only in response to determining that said data request is a data read request.

4. The method according to Claim 3, said step of determining further comprising:

in response to determining that said data request is a data write request, logging said data write request;

storing new data for said data write request;

setting said particular storage medium to a rebuild status;

rebuilding said particular storage medium utilizing said data and related parity data from said other storage media; and

setting said particular storage medium to online status.

5. The method according to Claim 1, wherein said particular storage medium is a disk, and said redundant storage array is a redundant disk array and wherein:

receiving a data request for data stored by a particular storage medium comprises receiving a data request for data stored by a particular disk of said redundant disk array.

6. An adapter for use with a data processing system including a redundant storage array including a plurality of storage media containing data and related parity data, said adapter comprising:

an updating circuit for receiving a data request for data stored by a selected storage medium among said plurality of storage media; and

a request servicing circuit, in response to receiving said data request while an update to said selected storage medium is being performed, for servicing said data request by reference to said data and said related parity data of other storage media among said plurality of storage media.

7. The adapter according to Claim 6, wherein said updating circuit is an adapter processor.

8. The adapter according to Claim 6, wherein said request servicing circuit is an exclusive-OR (XOR) processor for automatically reconstructing requested data utilizing data and related parity data from other storage media during said updating, in response to a data request by said system processor, said XOR processor coupled to an adapter processor.

9. The adapter according to Claim 6, further including:

a non-volatile random access memory (NVRAM) for logging data write requests.

10. The adapter according to Claim 6, further comprising:

a storage location for temporarily storing write data for said selected storage media during said updating.

11. The adapter according to Claim 10, wherein said storage location is an adapter cache.

12. The adapter according to Claim 11, wherein said adapter cache further includes:

a battery-backup circuit, for preserving data stored on said adapter cache.

13. The adapter according the Claim 8, wherein said data request is a data read request.

14. A disk system comprising:

an adapter according to Claim 6; and

a redundant storage array.

15. The adapter according to Claim 6, wherein said redundant storage array is a redundant array of inexpensive disks (RAID).

16. A data processing system, comprising:

a system processor;

a system memory coupled to said system processor;

a redundant storage array; and

an adapter including a redundant storage array including a plurality of storage media containing data and related parity data, said adapter further including:

an updating circuit for receiving a data request for data stored by a selected storage medium among said plurality of storage media; and

a request servicing circuit, in response to receiving said data request while an update to said selected storage medium is being performed, for servicing said data request by reference to said data and said related parity data of other storage media among said plurality of storage media.

17. The data processing system according to Claim 16, wherein said redundant storage array is a redundant array of inexpensive disks (RAID).

18. The data processing system according to Claim 16, wherein said redundant storage array stores data in stripes, wherein each said stripe further includes data and related parity data.

19. The data processing system according to Claim 16, wherein said updating circuit is an adapter processor.

20. The data processing system according to Claim 18, wherein said request servicing circuit is an exclusive-OR (XOR) processor for automatically reconstructing requested data utilizing said data said related parity data from other storage media during said updating, in response to a data request by said system processor, said XOR processor coupled to said adapter processor.

21. The data processing system according to Claim 16, wherein said adapter further comprising:

a non-volatile random access memory (NVRAM) for logging all data write requests and data read-with-intent-to-write requests.

22. The data processing system according to Claim 16, further including:

a storage location for temporarily storing all write data for said selected storage media during said updating.

23. The data processing system according to Claim 22, wherein said storage location is a temporary space on other storage media of said redundant storage array.

24. The data processing system according to Claim 22, wherein said storage location is an adapter cache.

25. The data processing system according to Claim 24, wherein said adapter further includes a battery backup circuit.

26. A computer program product comprising:

a computer-readable medium;

updating code encoded within said computer-readable medium for receiving a data request for data stored by a selected storage medium among a plurality of storage media; and

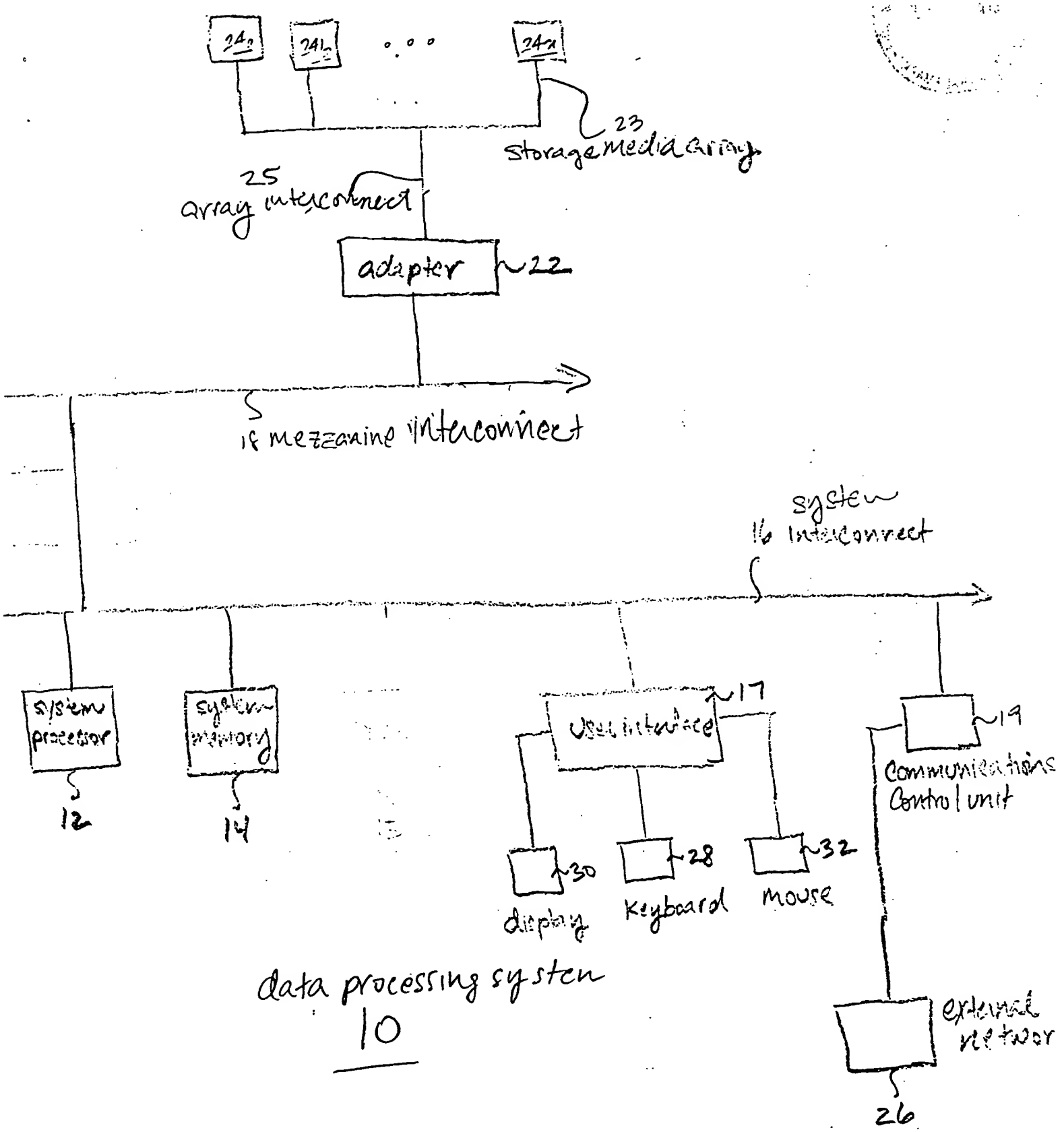
request servicing code encoded within said computer-readable medium, in response to receiving said data request while an update to said selected storage medium is being performed, for servicing said data request by reference to data and related parity data of other storage media among said plurality of storage media.

ABSTRACT OF THE DISCLOSURE

SYSTEM AND METHOD FOR SERVICING REQUESTS TO A STORAGE ARRAY

According to the method of the present invention, if a data read request to an unavailable particular storage medium is received from a system processor, a request servicing circuit of an adapter reconstructs the request data from data and parity data stored on other storage media in the array. However, if a data write request to an unavailable storage medium is received from the system processor, the actual data related to the data write request is stored in a temporary memory location, and the data write request is logged in a non-volatile memory. There is no built-in data redundancy in a first preferred embodiment of the present invention; thus, modified data is lost in the event of the failure of another storage medium during the update of the selected storage medium. However, a second preferred embodiment of the present invention incorporates a spare storage medium that preserves the modified data if another storage medium fails.

FIGURE 1



data processing system
10

FIGURE 2

RAID Adapter Block Diagram

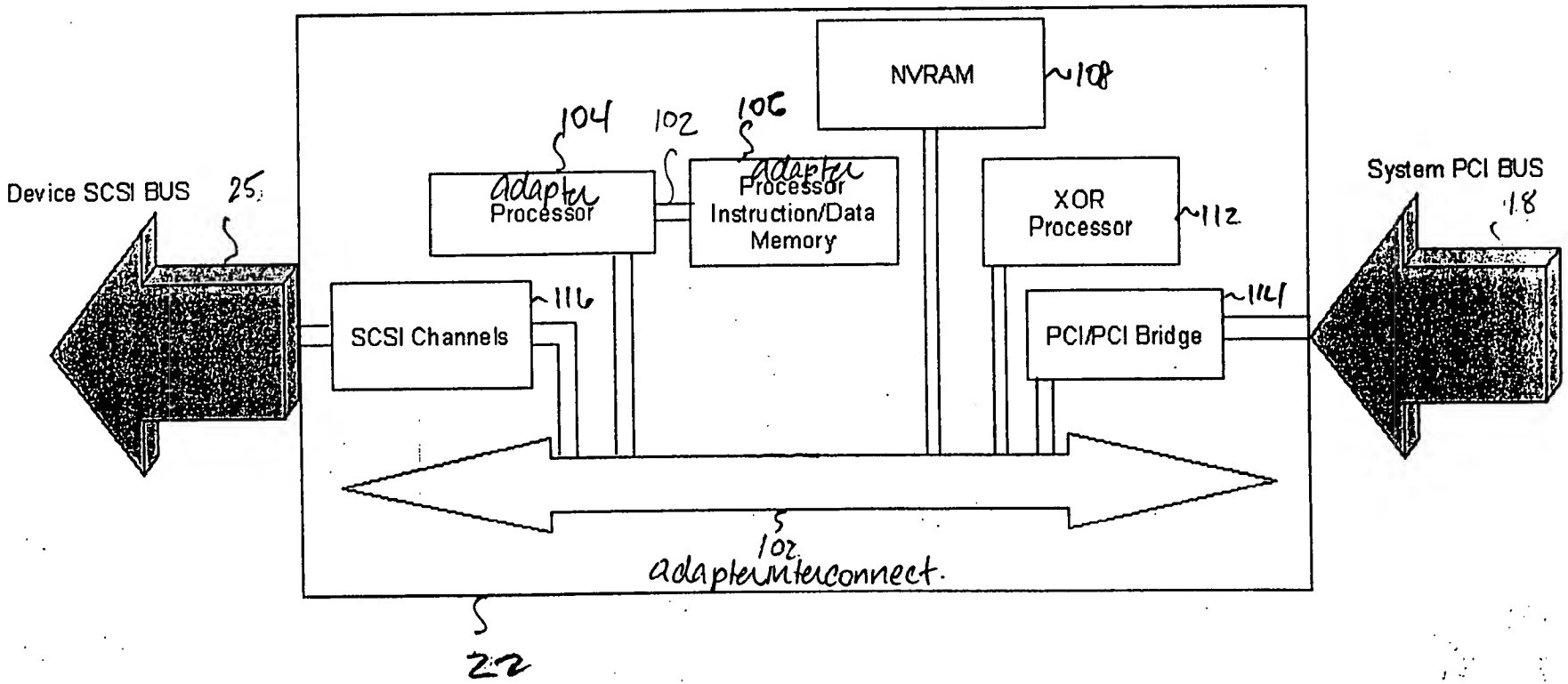
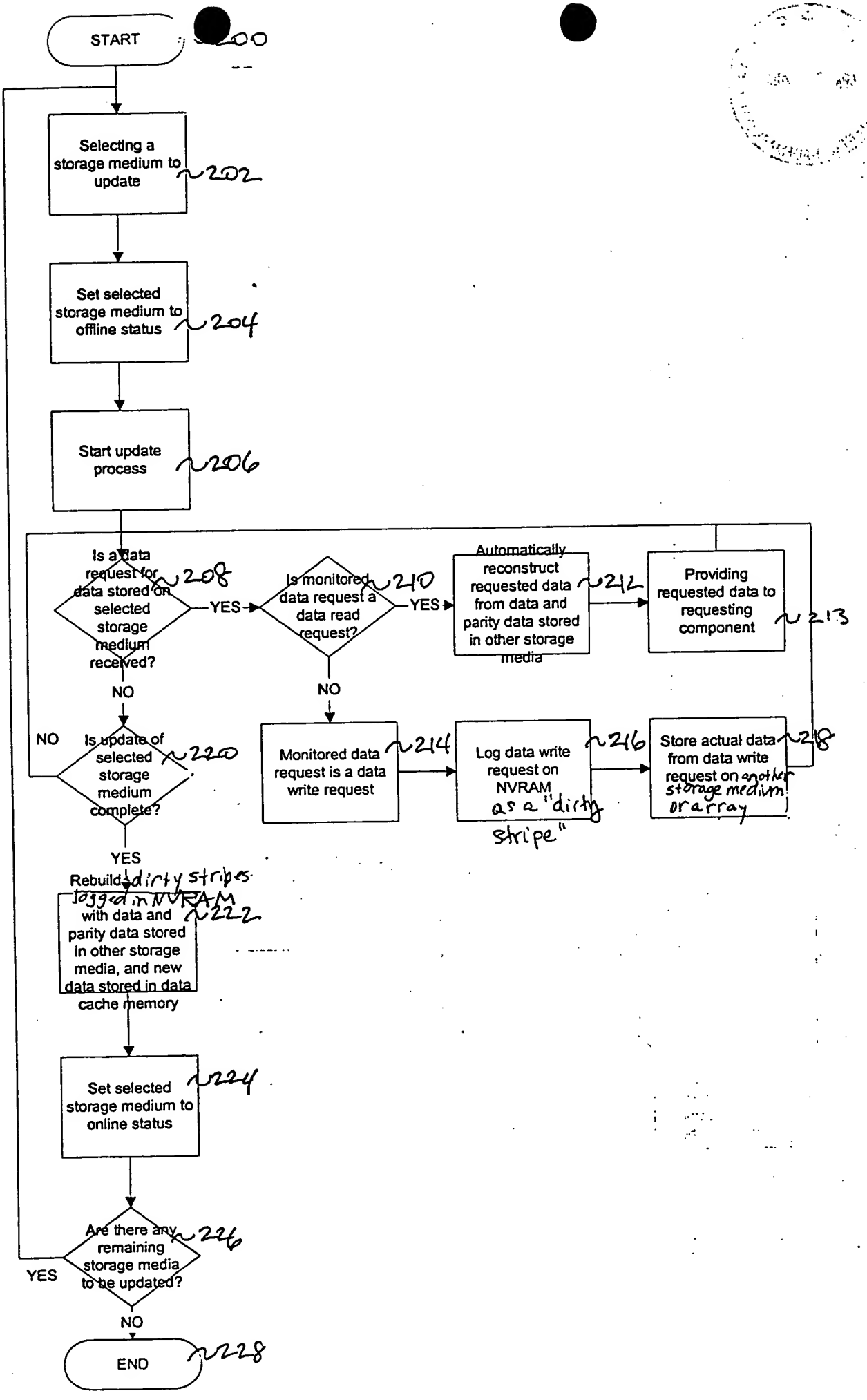


FIGURE 3A


START
200
Selecting a storage medium to update
202
Set selected storage medium to offline status
204
Start update process
206
Is a data request for data stored on selected storage medium received?
208
YES
Is monitored data request a data read request?
210
YES
Automatically reconstruct requested data from data and parity data stored in other storage media
212
Providing requested data to requesting component
213
NO
NO
Is update of selected storage medium complete?
220
Monitored data request is a data write request
214
Log data write request on NVRAM as a "dirty stripe"
216
Store actual data from data write request on another storage medium or array
218
YES
Rebuild dirty stripes logged in NVRAM with data and parity data stored in other storage media, and new data stored in data cache memory
222
Set selected storage medium to online status
224
Are there any remaining storage media to be updated?
226
YES
NO
END
228

IGURE 3B

(updating)

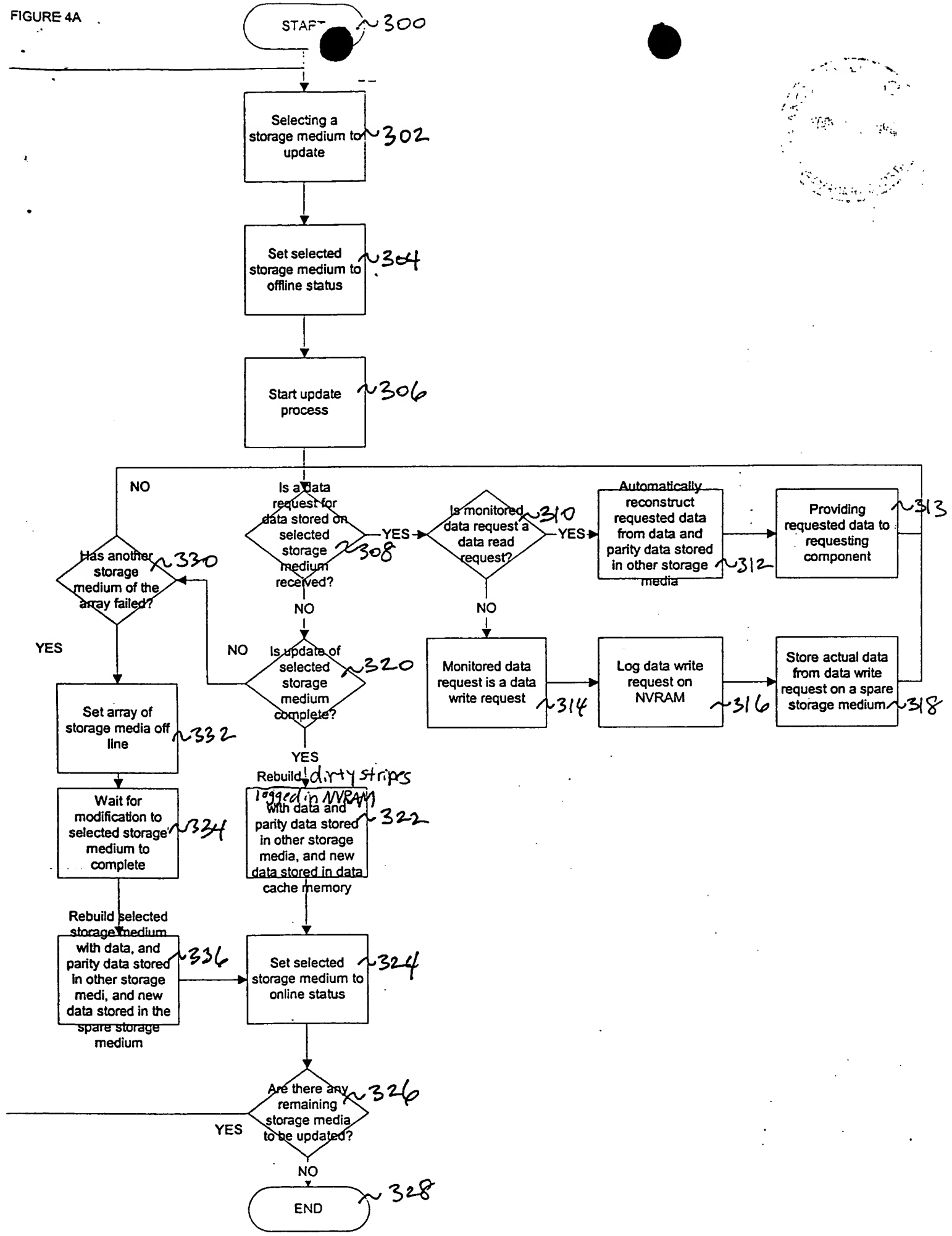
STRIPE NUMBER	STORAGE MEDIUM 1	STORAGE MEDIUM 2	STORAGE MEDIUM 3
1	D0	D1	P1
2	D2	P2	D3
3	P3	D4	D5
...	...	...	...
N	DN	PN	DN+1

IGURE 4B

(updating)

STRIPE NUMBER	STORAGE MEDIUM 1	STORAGE MEDIUM 2	STORAGE MEDIUM 3	SPARE STORAGE MEDIUM 4
1	D0	P1	P1	D1'
2	D2	P2	D3	
3	P3	D4	D5	
...	...	...	...	...
N	DN	PN	DN+1	

FIGURE 4A


START
300
Selecting a storage medium to update
302
Set selected storage medium to offline status
304
Start update process
306
Is a data request for data stored on selected storage medium received?
308
YES
Is monitored data request a data read request?
310
YES
Automatically reconstruct requested data from data and parity data stored in other storage media
312
Providing requested data to requesting component
313
NO
NO
Has another storage medium of the array failed?
330
YES
NO
Is update of selected storage medium complete?
320
Monitored data request is a data write request
314
Log data write request on NVRAM
316
Store actual data from data write request on a spare storage medium
318
Set array of storage media off line
332
YES
Rebuild dirty stripes logged in NVRAM with data and parity data stored in other storage media, and new data stored in data cache memory
322
Wait for modification to selected storage medium to complete
334
Rebuild selected storage medium with data, and parity data stored in other storage medi, and new data stored in the spare storage medium
336
Set selected storage medium to online status
324
Are there any remaining storage media to be updated?
326
YES
NO
END
328

APR 2 2 2002

UNITED STATES PATENT AND TRADEMARK OFFICE

COMMISSIONER FOR PATENTS
UNITED STATES PATENT AND TRADEMARK OFFICE
WASHINGTON, D.C. 20231
www.uspto.gov

APPLICATION NUMBER	FILING/RECEIPT DATE	FIRST NAMED APPLICANT	ATTORNEY DOCKET NUMBER
09/894,077	06/28/2001	Dean Alan Kalman	RPS920000119US1

CONFIRMATION NO. 6529

28722
BRACEWELL & PATTERSON, L.L.P.
7600B NORTH CAPITAL OF TEXAS HIGHWAY
SUITE 350
AUSTIN, TX 78731-1168

FORMALITIES LETTER

OC000000007611473

COPY OF PAPERS
ORIGINALLY FILED

Date Mailed: 03/11/2002

NOTICE OF OMITTED ITEM(S) IN A NONPROVISIONAL APPLICATION

FILED UNDER 37 CFR 1.53(b)

A filing date has been accorded to the above-identified nonprovisional application papers; however, the following item(s) appear to have been omitted from the application:

- Figure(s) **3a,3b** described in the specification.

I. Should applicant contend that the above-noted omitted item(s) was in fact deposited in the U.S. Patent and Trademark Office (USPTO) with the nonprovisional application papers, a copy of this Notice and a petition (and $130.00 petition fee *(37 CFR 1.17(h))*) with evidence of such deposit **must** be filed within **TWO MONTHS** of the date of this Notice. The petition fee will be refunded if is determined that the item(s) was received by the USPTO.

II. Should applicant desire to supply the omitted item(s) and accept the date that such omitted item(s) was filed in the USPTO as the filing date of the above-identified application, a copy of this Notice, the omitted item(s) (with a supplemental oath or declaration in compliance with 37 CFR 1.63 and 1.64 referring to such items), and a petition under 37 CFR 1.182 (with the $130.00 petition fee *(37 CFR 1.17(h))* requesting the later filing date **must** be filed within **TWO MONTHS** of the date of this Notice.

III. The failure to file a petition (and petition fee) under the above options (I) or (II) within **TWO MONTHS** of the date of this Notice (37 CFR 1.181(f)) will be treated as a constructive acceptance by the applicant of the application as deposited in the USPTO. **THIS TWO MONTH PERIOD IS NOT EXTENDABLE UNDER 37 CFR 1.136(a) or (b)**. In the absence of a timely filed petition in reply to this Notice, the application will maintain a filing date as of the date of deposit of the application papers in the USPTO, and original application papers (*i.e.*, the original disclosure of the invention) will include only those application papers present in the USPTO on the date of deposit.

In the event that applicant elects not to take action pursuant to options (I) or (II) above (thereby constructively electing option (III)), amendment of the specification to renumber the pages consecutively and cancel incomplete sentences caused by any omitted page(s), and/or amendment of the specification to cancel all references to any omitted drawing(s), relabel the drawing figures to be numbered consecutively (if necessary), and correct the references in the specification to the drawing figures to correspond with any relabelled drawing figures, is required. Any drawing changes should be accompanied by a copy of the drawing figures showing the proposed changes in red ink. Such amendment and/or correction to the drawing figures, if necessary, should be by way of preliminary amendment submitted prior to the first Office action to avoid delays in the prosecution of the application.

A copy of this notice <u>MUST</u> be returned with the reply.

Alia Mushir

Customer Service Center
Initial Patent Examination Division (703) 308-1202
PART 2 - COPY TO BE RETURNED WITH RESPONSE